Refine Search

Search Results -

Terms	Documents
L9 and L1	12

Database:
US Pre-Grant Publication Full-Text Database
US Patents Full-Text Database
US OCR Full-Text Database
EPO Abstracts Database
JPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Search: L10

Refine Search

Recall Text | Clear | Interrupt

Search History

DATE: Wednesday, August 11, 2004 Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
DB=USPT; PLUR=YES; OP=OR			
L10	L9 and l1	12	L10
L9	L8 and l7	44	L9
L8	galois adj1 field adj1 adder	52	L8
L7	galois adj1 field adj1 multiplier	136	L7
L6	galois adj3 multipler	0	L6
L5	galois adj1 field adj1 multipler	0	L5
L4	galois adj field adj1 multipler	0	L4
L3	L2 and l1	0	L3
L2	6571368[pn]	1	L2
L1	bch adj1 code	807	L1

END OF SEARCH HISTORY

Refine Search

Search Results -

Terms	Documents
L10 same decod$	13

Database:

US Pre-Grant Publication Full-Text Database
US Patents Full-Text Database
US OCR Full-Text Database
EPO Abstracts Database
JPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Search: L14

Refine Search

Recall Text | Clear | Interrupt

Search History

DATE: Wednesday, August 11, 2004 Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
DB=USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR			
L14	L10 same decod$	13	L14
L13	L10 and l11	3	L13
L12	L11 same l10	0	L12
L11	error adj1 polynomial	238	L11
L10	L6 same l3	64	L10
L9	L6 and l3	68	L9
L8	L4 and l6	0	L8
L7	L6 and l5	0	L7
L6	galois adj1 field adj1 adder	76	L6
L5	multiplier adj1 feeding	95	L5
L4	multiply near2 accumulator	410	L4
L3	galois near3 multiplier	354	L3
DB=USPT; PLUR=YES; OP=OR			
L2	multiply near2 accumulator	330	L2

L1	galois near3 multiplier	182	L1

END OF SEARCH HISTORY

L13: Entry 2 of 3 File: USPT May 9, 1995

DOCUMENT-IDENTIFIER: US 5414719 A
TITLE: Operating circuit for galois field

Brief Summary Text (15):
The second Galois field operating circuit includes two Galois field multipliers and one Galois field adder. This Galois field operating circuit operates to receive four elements X1, X2, Y1 and Y2 of the Galois field, to perform an operation of Z=X1.times.X2+Y1.times.Y2, and feed the operational result Z. For example, to calculate a determination of two-dimensional matrix, the following operations can be taken.

Brief Summary Text (65):
The term coefficients .omega..sub.0, .omega..sub., ...omega..sub.n-1 of the error evaluator polynomial .omega. (z) are obtained by the following relation of ##EQU8## In a case of performing an erasure plus error correction, since the error number m and the error locator X.sub.i (h+1.ltoreq.i.ltoreq.n) are unknown it is necessary to derive those values in advance. For this purpose, it is necessary to derive the error locator polynomial .sigma. (z) dedicated to the error and a polynomial T(z) with respect to z meeting the following expression, ##EQU9##

L13: Entry 3 of 3 File: USPT Oct 10, 1989

DOCUMENT-IDENTIFIER: US 4873688 A
**** See image for Certificate of Correction ****
TITLE: High-speed real-time Reed-Solomon decoder

Detailed Description Text (23):
During transmission the code suffers changes due to noise in the channel which amounts to an error polynomial being added to the code polynomial C(x). Let the received polynomial be

Detailed Description Text (24):
where E(x) is the error polynomial and each r.sub.i is a field element. The first step in the decoding algorithm is to calculate the syndromes. R(x) is evaluated at each of the zeros of G(x), i.e., at .beta..sup.s, .beta..sup.s+1, .beta..sup.s+2 . . . , .beta..sup.s+2t-1. Because the code word, C(x) is a multiple of G(x) the zeros of G(x) are also zeros of C(x). For each of the 2t roots of G(x), the Code Polynomial C(x) is zero. Thus, the syndromes are r(.alpha..sup.0)=e (.alpha..sup.1) . . . , r(.alpha..sup.2T-1)=e(.alpha..sup.2T.sup.-1). The 2t syndromes are ##EQU3## where 0.ltoreq.k.ltoreq.2t-1. The value of the syndrome polynomial can be defined as ##EQU4## Another equivalent definition of the syndrome polynomial is as follows. The syndrome polynomial is defined as

Detailed Description Text (99):
The Euclid divide module is comprised of 2t cells, 0 through 2t-1. Each of the cells 0 through 2t-2 is identical. Referring to cell 0, a typical cell each cell has an M.sub.i register 102, an M.sub.i-1 register 104, a multiplexer 106, a general Galois Field multiplier 108 and a Galois Field adder 110. Each of the registers M.sub.i and M.sub.i-1 are eight bit registers. The M.sub.i register 102 is coupled to receive an input. The output of the M.sub.i register is coupled to a first input of the multiplexer 106. The multiplexer 106 has a first input, a second input, a first output and a second output. The multiplexer may be configured in one of two ways as shown in FIG. 8; cycle 1 and cycle 2. The first configuration of the multiplexer passes a signal through from the first input to the first output and similarly from the second input to the second output. The second configuration of the multiplexer passes a signal from the first input to the second output and from the second input to the first output as shown in FIG. 8 cycle 2 for cell zero. The first output of the multiplexer 106 is coupled to a first input of the general multiplier 108 and as an input to the register M.sub.i-1 104. The output of the multiplier 108 is coupled to the first input of the adder 110. The second output of the multiplexer 106 is coupled to a second input of the adder 110. The output of the adder 110 is coupled to the input of the M.sub.i register of the next cell 112. The multiplier 108 is also coupled to receive the quotient Q(x) as a multiplicand on the second input.

Detailed Description Text (109):
The Euclid multiply algorithm is performed by the circuit shown in the block diagram of FIG. 9. Each of the t+1 cells, cells 0 through t are designed the same. For example cell t is comprised of a multiplexer 200, a Galois Field multiplier 202, a Galois Field adder 204, an L.sub.i register 206 and a L.sub.i-1 register

208. Each of the registers L.sub.i and L.sub.i-1 are 8 bit bytes. The multiplexer 200 is the same as the multiplexers in the Euclid divide circuit. In cell t the first input of the multiplexer 200 is coupled to receive the output of register L.sub.i.sbsb.t 206. The second input of multiplexer 200 is coupled to receive a zero. The first output of the multiplexer 200 is coupled to a first input of the adder 204.

CLAIMS:

4. The Galois Field polynomial solver according to claim 3 wherein the improvement further comprises evaluating each polynomial for a highest order coefficient of an error polynomial first to a lowest order coefficient last.

Previous Doc Next Doc Go to Doc#

L13: Entry 1 of 3 File: USPT May 27, 2003

DOCUMENT-IDENTIFIER: US 6571368 B1
TITLE: Systolic Reed-Solomon decoder

Brief Summary Text (6):
Reed-Solomon decoding generally involves four steps. In the first two steps, a syndrome polynomal S(x) is generated and the key equation .LAMBDA.(x)S(x)=.OMEGA.(x) mod x.sup.2t is solved to obtain an error location polynomial .LAMBDA.(x) and an error evaluator polynomial .OMEGA.(x). Step three is to evaluate these polynomials to determine which symbols are affected by errors and what are the error values, resulting in an error polynomial E(x). Finally, the error polynomial is combined with the received polynomial R(x) (which is buffered during steps one to three) to produce a reconstructed message without errors.

Detailed Description Text (9):
When a code word is transmitted, it is prone to pick up errors as a result of noisy communication channels, such as radio signals or copper telephone lines. A received polynomial R(x) may differ from C(x) because of noise. This code word corruption by noise can be viewed as adding an error polynomial ##EQU3##

Detailed Description Text (29):
Returning to FIG. 5, the block diagram shows the structure of a decoder embodying the principles of the present invention. The received message polynomial, individual terms denoted r.sub.j, is serially input to both a delay buffer 501 and syndrome calculation means 511. The delay buffer 501 holds one or more code words as an error polynomial E(x) is calculated for each received code word. After an appropriate delay, the polynomial code word is combined with E(x) at Galois Field adder 516 to produce the corrected message c(x). In carrying out the present invention, a conventional means for calculation of syndromes can be used. .LAMBDA. variation of Berlekamp's Power Sums Tower, Chapter 10 at p. 213 et seq. might be used. The general approach to calculating syndromes is described by Lin & Costello at p. 167-68 and 173-74. A detailed description of a syndrome calculator appears in Alok Sharma, "Methods and Apparatus for Error Correction", U.S. Pat. No. 5,889,793, FIG. 4 & col. 10 line 44 to column 13 line 26 (issued Mar. 30, 1999). A systolic array for syndrome calculation is described in Keiichi Iwamura, Yasunori Dohi & Hideki Imai, "A Design of Reed-Solomon Decoder with Systolic-Array Structure," IEEE Transactions on Computers, Vol. 44, No. 1, pp. 118-122 (January 1995). Alternatively, the specific structure for calculating syndromes described below can be used.

Detailed Description Text (30):
The output of the syndrome computation 511 is a syndrome polynomial S(x), preferably transferred by a parallel output into the Euclid's algorithm logic 512. The output of the Euclid's algorithm logic 512 is two polynomials, the error location polynomial .LAMBDA.(x), which is passed to evaluator. 513 and the error evaluator polynomial .OMEGA.(x), which is passed to evaluator 514. Error calculator 515 receives from evaluator 514 the result of evaluating .LAMBDA.(.alpha..sup.-1), at least when the result is zero, which identifies an error location in the input polynomial. Error calculator 515 receives from evaluator 513 the value .LAMBDA.'(.alpha..sup.-1) and receives from evaluator 514 the value .OMEGA.(.alpha..sup.-1), allowing calculation of error polynomial E(x) as indicated.

Galois Field adder 516, preferably a modulo-2 adder, combines E(x) and R(x) to produce the corrected message c(x).

Detailed Description Text (67):
FIGS. 24 and 25 depict how outputs of the lowest order cells in each of the three sub arrays are combined to generate the error location and error evaluator polynomials. In FIG. 24, the inputs to Galois Field multiplier 2401 are the output from the lowest order .LAMBDA.odd cell and the factor value at which .LAMBDA.odd is evaluated. The output of multiplier 2401 is combined with output from the lowest order .LAMBDA.even cell by Galois Field adder 2402. This output is an evaluation of the error location polynomial at a particular location. When this output is "0", the error value for this position is calculated by Galois Field divider 2501 as depicted in FIG. 25.

Detailed Description Text (68):
FIG. 26 depicts one control logic for the evaluation computation. As with FIGS. 8 and 13, numeric sequences appear in square brackets adjacent to the logic blocks of this figure. These bracketed numbers correspond to the time line numbers along the top of FIG. 32. In FIG. 26, the process begins in block 2601 with a test of the control signal ToEval. If the control signal is false, the process waits. When the signal is true, control passes to block 2602. This block represents enabling control signal EnXi, which initiates the process of evaluating .OMEGA.(x). From block 2602, processing proceeds along two paths. Because the even and odd components of .LAMBDA.(x) are calculated in parallel, the evaluation of .OMEGA.(x) begins before the evaluation of .LAMBDA.(x). Along one path, the process flows from 2602 to delay element 2603. Enablement of control signal EnX.sub.2i is delayed so that the results of evaluating .OMEGA.(x) and .LAMBDA.(x) will be generated the same time. After the delay in 2603, the second control signal EnX.sub.2i is enabled in block 2604. Processing proceeds in parallel along two paths. These paths from block 2602 and through block 2604 converge at block 2605, which indicates that the control signals remain enabled for a predetermined number of cycles. From block 2605, the process proceeds to block 2606 where terms of the error polynomial E(x) are sequentially output in N clock cycles.

Detailed Description Text (71):
Assume the message polynomials is M(x)=.alpha..sup.5 x.sup.2 +.alpha..sup.3 x+.alpha..sup.6. The encoded code word polynomial is C(x)=.alpha..sup.5 x.sup.6 +.alpha..sup.3 x.sup.5 +.alpha..sup.6 x.sup.4 +.alpha..sup.5 x.sup.3 +x.sup.2 +x+.alpha..sup.3. Suppose that two errors occur as the error polynomial E(x) =.alpha..sup.2 x.sup.4 +1. Then the received polynomial is R(x)=.alpha..sup.5 x.sup.6 +.alpha..sup.3 x.sup.5 +x.sup.4 +.alpha..sup.5 x.sup.3 +x.sup.2 +x+.alpha..

Detailed Description Text (78):
In FIG. 32, from sequence from time 19 to time 29, is same with those in FIG. 28, processing the first received code word. From time 29 to time 39, FIG. 32 shows the processing of the second received code word. The error polynomial E(x) is correctly computed, as can be seen by comparing the result of the simulation to the assumed error term.

Previous Doc Next Doc Go to Doc#

L14: Entry 2 of 13 File: USPT Jun 15, 1999

DOCUMENT-IDENTIFIER: US 5912905 A

**** See image for Certificate of Correction ****

TITLE: Error-correcting encoder, error-correcting decoder and data transmitting system with error-correcting codes

Abstract Text (1):

The present invention provides an error-correcting encoder and an error-correcting decoder which encode/decode a plurality of information symbols in parallel with a reduced number of shifts, which enables a reduction in the processing time. The error-correcting encoder of the invention includes a shift register including stages equal to a predetermined number of check symbols for inputting different information symbols in parallel from a plurality of input terminals. The encoder also includes a Galois field multiplier for multiplying each coefficient and a Galois field adder to obtain the predetermined number of check symbols from the information symbols. The encoder can generate the predetermined number of check symbols with shifts, the number of which is reduced according to the number of parallel inputs. The syndrome generator of the error-correcting decoder of the invention includes a plurality of Galois field multipliers which multiply the coefficients for calculating syndromes for inputting different code symbols in parallel from a plurality of input terminals. The syndrome generator also includes a Galois field adder and a shift register to obtain the predetermined syndrome generating polynomial. The syndrome generator can obtain the desired syndromes with shifts, the number of which is reduced according to the number of parallel inputs.

CLAIMS:

13. An error-correcting system for encoding information symbols and decoding code symbols, comprising:

(A) an error-correcting encoder for generating check symbols, and for generating code symbols, wherein the error-correcting encoder has plural shift-registers including stages equal to a number of check symbols, for generating check symbols with a reduced number of shifts, and has plural Galois field adders and plural Galois field multipliers for forming check symbols generating logic so that same check symbols are produced with the reduced number of shifts using plural information symbols at a time and output of the shift-registers; and

(C) an error-correcting decoder for decoding code symbols, wherein the error-correcting decoder has plural syndrome generators including plural Galois field multipliers, and including plural Galois field adders, for generating syndromes of received words by using plural received words with a reduced number of shifts, and has a Chien-Search circuit for analyzing error locations by using the coefficients of error-location polynomials.

L14: Entry 5 of 13 File: USPT Dec 16, 1997

DOCUMENT-IDENTIFIER: US 5699368 A
TITLE: Error-correcting encoder, error-correcting decoder, and data transmitting system with error-correcting codes

Abstract Text (1):
The present invention provides an error-correcting encoder and an error-correcting decoder which encode/decode a plurality of information symbols in parallel with a reduced number of shifts, which enables a reduction in the processing time. The error-correcting encoder of the invention includes a shift-register including stages equal to a predetermined number of check symbols for inputting different information symbols in parallel from a plurality of input terminals. The encoder also includes a Galois field multiplier for multiplying each coefficient and a Galois field adder to obtain the predetermined number of check symbols from the information symbols. The encoder can generate the predetermined number of check symbols with shifts, the number of which is reduced according to the number of parallel inputs. The syndrome generator of the error-correcting decoder of the invention includes a plurality of Galois field multipliers which multiply the coefficients for calculating syndromes for inputting different code symbols in parallel from a plurality of input terminals. The syndrome generator also includes a Galois field adder and a shift-register(s) to obtain the predetermined syndrome generating polynomial. The syndrome generator can obtain the desired syndromes with shifts, the number of which is reduced according to the number of parallel inputs.

CLAIMS:

4. An error-correcting system for encoding information symbols and decoding code symbols, comprising:

(A) an error-correcting encoder for generating check symbols, and for generating code symbols, wherein the error-correcting encoder has a shift register including stages, S.sub.a -S.sub.d, equal to a number of check symbols, for generating check symbols in a reduced number of shifts, and has plural Galois field adders and plural Galois field multipliers for forming check symbols generating logic configured with the shift register and a plurality of EXOR gates to compute and store in the plurality of stages

S.sub.a =(g.sub.3.sup.2 +g.sub.2)A+g.sub.3 B+C+(g.sub.3.sup.2 +g.sub.2)m.sub.j +g.sub.3 m.sub.i

S.sub.b =(g.sub.3 g.sub.2 +g.sub.1)A+g.sub.2 B+D+(g.sub.3 g.sub.2 +g.sub.1)m.sub.j +g.sub.2 m.sub.i

S.sub.c =(g.sub.3 g.sub.1 +g.sub.0)A+g.sub.1 B+(g.sub.3 g.sub.1 +g.sub.0)m.sub.j +g.sub.1 m.sub.i

S.sub.d =(g.sub.3 g.sub.0)A+g.sub.0 B+(g.sub.3 g.sub.0)m.sub.j +g.sub.0 m.sub.i,

where

A, B, C and D are previous contents of S.sub.a -S.sub.d, respectively, g.sub.0 -

g.sub.3 are Galois Field constants and m.sub.i and m.sub.j are information symbols,

so that said check symbols are produced in the reduced number of shifts using plural information symbols at a time and output of the shift-registers; and

(C) an error-correcting decoder for decoding code symbols, wherein the error-correcting decoder has plural syndrome generators including plural Galois field multipliers and including plural Galois field adders, for generating syndromes of a received word by using plural received symbols with a reduced number of shifts, and has a Chien-Search circuit for analyzing error locations by using the coefficients of an error-location polynomial.

Previous Doc Next Doc Go to Doc#

First Hit Previous Doc Next Doc Go to Doc#

End of Result Set

Generate Collection Print

L14: Entry 13 of 13 File: DWPI Sep 28, 1995

DERWENT-ACC-NO: 1995-337939
DERWENT-WEEK: 200240

TITLE: Error-correcting encoder using Galois field adder and multipliers - includes logic for generating test symbols by min. number of shifts with use of several input signals of data symbols at same time

Basic Abstract Text (2):
The shift registers have stages equal in number to the test symbols for parallel input of different data symbols. The syndrome generator of the decoder also contains Galois field multipliers, a Galois field adder and a shift register for the predetermined syndrome generation polynomial.

Equivalent Abstract Text (2):
The shift registers have stages equal in number to the test symbols for parallel input of different data symbols. The syndrome generator of the decoder also contains Galois field multipliers, a Galois field adder and a shift register for the predetermined syndrome generation polynomial.

Equivalent Abstract Text (5):
The shift registers have stages equal in number to the test symbols for parallel input of different data symbols. The syndrome generator of the decoder also contains Galois field multipliers, a Galois field adder and a shift register for the predetermined syndrome generation polynomial.

Previous Doc Next Doc Go to Doc#